REMARKS/ARGUMENTS

Claims 1-58 are pending in the application. Claims 30-58 are rejected under 35 U.S.C. 112; and claims 1-58 are rejected as anticipated under 35 U.S.C. 102(b).

Claim Amendments

Amended independent claims 1 and 30 propose a method and system, respectively, for administration of network financial transaction terminals, in which an event query is sent to a management instrumentation application on one of the financial transaction terminals, and a pre-determined event type is subscribed to, by the queued component client, which thereafter captures and consumes an event message from the management instrumentation application before writing the event message into an event log on the financial transaction terminal and which sends the captured and consumed event message to a server site event queue via message queuing services components, whereupon the event message is removed from the server site event queue and stored into a database by a queued component server.

Claims 6, 7, 35, and 36 are canceled and claims 8 and 37 are amended to address editorial issues raised by the amendment of claims 1 and 30, and claims 30-34, 37-53, and 58 are amended to address editorial issues raised by the Examiner's rejection of claims 30-58 as being indefinite. Support for the foregoing amendment is found throughout the specification and in the claims and as detailed above. Accordingly, no new matter has been added.

Claim Rejections - 35 U.S.C. § 112

Claims 30-58 stand rejected under 35 U.S.C. § 112 as indefinite because they claim a system, but the Examiner considers that it is unclear whether the system is an apparatus or a method. While the rejection is not believed to be appropriate, the foregoing amendment of claims 1 and 30 positively reciting components of the claimed system overcomes the rejection.

10

Claim Rejections - 35 U.S.C. § 102

Claims 1-58 stand rejected as anticipated by Brown (U.S. 5,857,190) under 35 U.S.C. § 102(b). The rejection is respectfully traversed and reconsideration is requested. The reference asserted does not read on the claimed invention. On the contrary, instead of a method and system for administration of network financial transaction terminals, as recited in independent claims 1 and 30, respectively, Brown discloses a system for centrally logging large numbers of concurrently received events, including channel up/down, button presses, service requests, warnings, and errors generated and detected at user interface units, such as TV sets controlled by set-top boxes of an interactive entertainment network, in subscriber homes . See, e.g., Brown, Col 2, lines 37-55; Col 3, line 65-Col 4, line 7.

There is no teaching or suggestion whatsoever in Brown of a management instrumentation application or a queued component client on a financial transaction terminal, nor of sending an event query to the management instrumentation application and subscribing to a pre-determined event type by the queued component client on the financial transaction terminal, nor of receiving an event notification from the management instrumentation application by the queued component client, as recited in claims 1 and 30, respectively. On the contrary, according to Brown, a user interface unit processor simply detects events by tracking when a viewer depresses keys or enters a wrong password and by using interrupts. See, e.g., Brown, Col 6, lines 25-33.

Moreover, Brown fails to teach or suggest a queued component client that captures and consumes an event message before writing the event message into an event log on the financial transaction terminal and that sends the captured and consumed event message to a server site event queue via message queuing services components, as recited in amended claims 1 and 30, respectively. Instead, according to Brown, upon receiving event-related interrupts (i.e., channel selections) by the user interface unit processor, an event evaluator 52 looks to an event filter criteria 54 to determine whether an event at the user interface unit is loggable, and if so, simply records the loggable event. See, e.g., Brown, Col 6, line 65-Col 7, line 14. Further, according to Brown, the event evaluator 52 at the user interface unit 26 then reports loggable events over the network 28 to an event

11

log manager 56 at a headend 22 that manages where events are logged. <u>See</u>, e.g., Brown, Col 7, lines 15-25.

In other words, instead of capturing and consuming an event message by the queued component client before writing the event message into an event log on the financial transaction terminal and sending the captured and consumed event message to a server site event queue by the queued component client, as recited in amended claims 1 and 30, respectively, according to Brown, events are initially captured at the processor of the user interface unit and the event evaluator then determines whether the event is a loggable by looking to the event filter criteria 54 at the user interface unit. <u>See</u>, e.g., Brown, Col 9, line 63-Col 10, line 13. Thus, Brown teaches away from Applicants' claimed invention, a key feature of which is the capture of event messages in real time as it occurs, prior to writing any event message content to the event log on the transaction terminal, so that there is absolutely no chance for data tampering at the client site. <u>See</u>, e.g., Specification, p. 11, lines 4-16.

Neither does Brown teach or suggest removing the event message from the server site event queue by a queued component server and storing the event message into a database by the queued component server. Rather, according to Brown, an object at the headend selects the appropriate database to store the event information using a service string stored at the headend with a relational list correlating the appropriate database for a given kind of event reported by a log API. <u>See</u>, e.g., Brown, Col 3, lines 22-31.

Consequently, Brown fails to disclose or even suggest the required combination of limitations of independent claims 1 and 30 proposing that an event query is sent to a management instrumentation application on one of the financial transaction terminals, and a pre-determined event type is subscribed to, by the queued component client, which thereafter captures and consumes an event message from the management instrumentation application before writing the event message into an event log on the financial transaction terminal and which sends the captured and consumed event message to a server site event queue via message queuing services components, and that the event

message is removed from the server site event queue and stored into a database by a queued component server, as recited in amended independent claims 1 and 30.
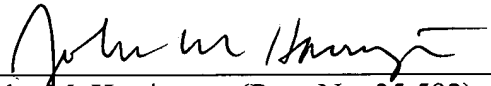
Because each and every element as set forth in amended claims 1 and 30 is not found, either expressly or inherently in Brown, the Examiner has failed to establish the required *prima facie* case of unpatentability. See Verdegaal Bros. v. Union Oil Co. of California, 814 F.2d 628 (Fed. Cir. 1987); See also MPEP §2131. The Examiner has failed to establish the required *prima facie* case of unpatentability for amended independent claims 1 and 30, and similarly has failed to establish a *prima facie* case of unpatentability for claims 2-5 and 8-29 (claims 6 and 7 being canceled) that depend on claim 1 and claims 31-34 and 37-58 (claims 38 and 39 being likewise canceled) that depend on claim 30, and which recite further specific elements that have no reasonable correspondence with the references.

Conclusion

In view of the foregoing amendment and these remarks, each of the claims remaining in the application is in condition for immediate allowance. Accordingly, the examiner is requested to reconsider and withdraw the rejection and to pass the application to issue. The examiner is respectfully invited to telephone the undersigned at (336) 607-7318 to discuss any questions relating to the application.

Respectfully submitted,

Date: 1/7/05

John M. Harrington (Reg. No. 25,592)
for George T. Marcou (Reg. No. 33,014)

Kilpatrick Stockton LLP
607 14th Street, NW, Suite 900
Washington, DC 20005
(202) 508-5800